

Article

Traffic Engineering Queue Optimization Models with Guaranteed Quality of Service Support

Larysa Titarenko ^{1,2}, Oleksandr Lemeshko ², Oleksandra Yeremenko ^{2,*}, Roman Savchenko ²
and Alexander Barkalov ¹

¹ Institute of Metrology, Electronics and Computer Science, University of Zielona Góra, ul. Licealna 9, 65-417 Zielona Góra, Poland; l.titarenko@imei.uz.zgora.pl (L.T.); a.barkalov@imei.uz.zgora.pl (A.B.)
² V.V. Popovskyy Department of Infocommunication Engineering, Kharkiv National University of Radio Electronics, Nauky Ave. 14, 61166 Kharkiv, Ukraine; oleksandr.lemeshko@nure.ua (O.L.); roman.savchenko1@nure.ua (R.S.)
* Correspondence: oleksandra.yeremenko@nure.ua

Abstract

The article introduces the Guarantee-Based Bandwidth Traffic Engineering Queue (GB(Bw)-TEQ) and Guarantee-Based Utilization Traffic Engineering Queue (GB(U)-TEQ) models for queue management on router interfaces. These models implement the principles of Traffic Engineering Queues and support both DiffServ and IntServ. Their novelty lies in the ability to provide guarantees either for the bandwidth allocated to a class queue or for its utilization coefficient. Such guarantees stabilize and control the average queue length, positively affecting key Quality of Service (QoS) indicators, particularly average delay and packet loss probability. The unreserved portion of the interface bandwidth is allocated among queues in proportion to their classes. Therefore, the higher-priority queues have lower utilization, while lower-priority queues operate with higher utilization, which is consistent with DiffServ principles. The models are formulated as a mixed-integer linear programming problem with an optimality criterion and a system of constraints. Computational experiments confirmed the operability and efficiency of GB(Bw)-TEQ and GB(U)-TEQ compared to the known analogue CB-TEQ model, which does not provide service-level guarantees. The results demonstrate that the proposed models achieve the stated guarantees and enable differentiated service without blocking the lowest-class queues. These solutions can be applied to automate queue management in IP/MPLS switches and routers as well as in software-defined networks.

Keywords: quality of service; traffic engineering; queue; optimization; resource allocation; classification; bandwidth; utilization



Academic Editor: Myung-Sup Kim

Received: 15 September 2025

Revised: 7 October 2025

Accepted: 14 October 2025

Published: 17 October 2025

Citation: Titarenko, L.; Lemeshko, O.; Yeremenko, O.; Savchenko, R.; Barkalov, A. Traffic Engineering Queue Optimization Models with Guaranteed Quality of Service Support. *Electronics* **2025**, *14*, 4078. <https://doi.org/10.3390/electronics14204078>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The current development of information and communication technologies has led to more complex network structures and higher demands on their efficient operation [1,2]. This makes it necessary to investigate mechanisms that can ensure stable and predictable data transmission. In this context, ensuring Quality of Service (QoS) is a central task in modern information and communication network engineering [3–5]. This task is inherently multifaceted and is complicated by several factors (Figure 1):

- wide geographical distribution of network elements;
- complex topologies;

- heterogeneity of technologies and protocols;
- highly dynamic network states due to variations in topology, load, and traffic characteristics;
- continuous growth in user number and QoS requirements;
- expansion and diversification of offered services;
- limited resources, including bandwidth, buffer capacity, and computational power of network and server equipment;
- coexistence of multiple QoS models (guaranteed, differentiated, best-effort);
- need for simultaneous control of multiple QoS metrics (bandwidth, delay, jitter, packet loss).



Figure 1. Factors Affecting the Complexity of QoS Provisioning.

The factors listed above can be addressed through two main approaches to ensuring QoS: extensive and intensive. The extensive approach is typically applied during network design, modernization, or restructuring. It relies on the addition of new network resources, such as deploying new segments, using more powerful communication devices and controllers, and implementing high-performance technologies at the physical and data link layers of the OSI model.

The intensive approach, in contrast, focuses on optimizing the use of existing resources through their formation, allocation, and reservation. In practice, both approaches are applied in combination, depending on the network's operating conditions, the competitive environment, and the availability of financial resources.

More generally, the problem of ensuring QoS requires the systematic and coordinated operation of technologies, protocols, and mechanisms across all OSI layers from the physical to the application layer. Each layer contributes in its own way to resource formation, distribution, and redistribution, as well as management, reservation, and recovery. Within this hierarchy, processes at the network layer play a significant role in maintaining a high level of QoS. The main mechanisms used at the network layer include (Figure 2) [6–10]:

- packet classification and marking (prioritization);
- traffic shaping and policing;
- routing protocols;
- signaling and reservation protocols for network resources;
- scheduling, resource allocation, and congestion management.

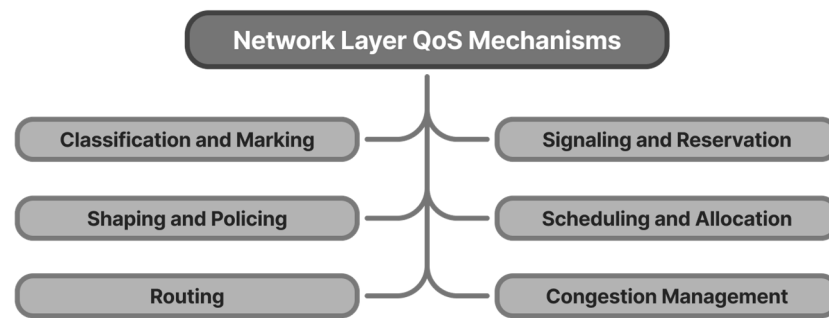


Figure 2. QoS Mechanisms at the OSI Network Layer.

These tools share a common objective: ensuring QoS. To achieve this, they must operate in a highly coordinated and harmonized manner. However, in practice, this objective has not been fully implemented. Although most of these tools are integrated into the hardware and software of the same communication devices or controllers, they often function independently. This autonomy typically reduces the efficiency of utilizing available network resources and, consequently, diminishes the overall level of QoS.

The principal mechanisms for distributing network resources in switches and routers are scheduling and resource allocation (Figure 2). These mechanisms are designed to establish and manage packet queues, as well as redistribute the bandwidth of network interfaces among them. They represent the core technological instruments for implementing the architectural models used to ensure QoS in IP networks, namely Integrated Services (IntServ) and Differentiated Services (DiffServ) [11,12].

Traditionally, scheduling and resource allocation mechanisms are expected to satisfy the following requirements [6–8]:

- support for multiple queues to enable service differentiation;
- provision of service guarantees through the reservation of link and buffer resources for specific traffic types;
- uniform processing of packets with identical characteristics (e.g., length, priority, class);
- high flexibility and a significant degree of automation in configuration;
- simplicity of algorithmic design and feasibility of software and hardware implementation in practice.

These requirements are often contradictory, making it practically impossible to satisfy all of them simultaneously. For instance, supporting a larger number of queues and enabling automated configuration inevitably increases the complexity of management algorithms, requiring the development of software that demands greater computational resources. Consequently, IP routers typically implement several scheduling and resource allocation mechanisms in parallel [6,7]. The choice of a particular mechanism depends on factors such as the current network state (e.g., load and QoS requirements) as well as the administrator's level of expertise, knowledge, and ability to configure the selected mechanism effectively.

The effectiveness of technological solutions for optimizing queue management on router interfaces largely depends on the efficiency of the mathematical models and methods that form the basis of the corresponding mechanisms. The practical application of such mathematical models and methods should automate the process of queue management and QoS assurance to minimize administrator intervention in these real-time processes, which are critical for the network. Therefore, the purpose of this article is to develop and enhance optimization models for queue management that support both guaranteed and differentiated quality of service in network routers. This goal is pursued through the following research objectives:

- conducting a comparative analysis of existing technological and theoretical solutions for queue management on router interfaces;
- improving and investigating optimization models of queue management to verify the adequacy and effectiveness of the proposed solutions;
- developing recommendations for practical implementation.

2. Overview of Queue Scheduling Mechanisms in Routers

Based on the analysis of the central scheduling and resource allocation mechanisms used in IP networks [6–9], a comparison was conducted according to key characteristics, including their operating principles, the number of supported queues, the degree of automation in management decisions, and their respective advantages and disadvantages (Table 1). This comparison enables the formulation of recommendations for their practical application and the identification of potential areas for improvement.

Table 1. Comparative Analysis of Queue Scheduling Mechanisms in Routers.

Name	Number of Queues	Operating Principle	Advantages	Disadvantages	Level of Automation
FIFO (First In, First Out)	1	Packets are processed in the order of arrival	Simple implementation, minimal resource consumption	No support for service differentiation	High
PQ (Priority Queuing)	4	Service order determined by queue priority	Minimal delay for high-priority traffic	Low-priority traffic may be blocked	Medium
CQ (Custom Queuing)	16	Service level of each queue regulated by byte counters	Allows differentiated servicing through configuration	Requires additional administrative configuration	Medium
WFQ (Weighted Fair Queuing)	16–4096	Service order depends on packet length and priority	Fair resource allocation	Large number of low-priority flows may reduce bandwidth for high-priority flows	High
CBQ (Class-Based Queuing)	64 class-based queues	Traffic is divided into classes with assigned bandwidth shares	Bandwidth control, service differentiation	Complex setup, processing overhead	Medium
CBWFQ (Class-Based Weighted Fair Queuing)	64 class-based queues, each configurable with WFQ	Combination of CBQ and WFQ	Provides QoS guarantees	Requires significant administrative configuration	Low
LLQ (Low Latency Queuing)	64 class-based queues, one designated as priority	Combination of CBWFQ with a priority queue	Minimal delay for critical traffic	Requires significant administrative configuration	Low

Among the analyzed solutions, the FIFO and WFQ mechanisms occupy a prominent position, as they operate automatically and require minimal administrative configuration. It should be noted that administrative intervention always causes significant delays in the control loop and requires a high level of qualification and experience from the network

administrator. Under low interface load conditions, the FIFO mechanism provides a high level of QoS without utilizing excessive computing resources of the switch/router processor. Under overload conditions, the use of FIFO is impractical. Therefore, it is appropriate to use WFQ, providing a high level of service differentiation based on the organization of 256 queues (by default), with the possibility of administratively increasing their number to 4096 [7]. However, with an increase in the number of low-priority packet flows, fairness in their service is accompanied by a redistribution of interface bandwidth in their favor, taking it away from high-priority packets. This can typically lead to a shortage of bandwidth needed for high-priority flows and, more importantly, a lack of QoS guarantees.

Another group of solutions, represented by CBQ, CBWFQ, and LLQ mechanisms, provides a high level of packet service differentiation and even QoS guarantees, as they support a relatively large number of queues. However, the CBQ mechanism is no longer used as a standalone solution on Cisco equipment. Instead, it functions as a component of more advanced mechanisms such as CBWFQ and LLQ.

CBWFQ and LLQ mechanisms create 64 class queues, each of which can be further configured with 16 to 4096 WFQ queues [7]. The high flexibility of these mechanisms is ensured by the fact that specific packet flows can be assigned to each class queue, and the required bandwidth and queue buffer size can be set. However, a significant drawback of CBWFQ and LLQ solutions is the high volume of manual settings that the administrator must perform. At the same time, the order of traffic classification, the value of the bandwidth, and the size of the queue buffer for each queue still need to be reasonably determined, which is practically impossible for administrators to do in real-time with such a large number of queues. This leads to the need for WFQ-like solutions that, in conditions of a large number of packet flows and queues, can automatically ensure both differentiated and guaranteed quality of service without significantly loading the computing resources of the switch or router. Solving this problem is impossible without developing or improving mathematical models and queue scheduling methods that serve as a mathematical and algorithmic basis for advanced technological solutions, ensuring the quality of service.

3. Novel Approaches to Queue Management in Networking

The demand for modern queue management (scheduling, congestion avoidance, congestion management, resource allocation) mechanisms is evident across diverse network types, as shown in the studies summarized in Table 2 [13–24]. In particular, challenges related to optimal resource allocation and QoS guarantees are critical not only in general-purpose telecommunications networks [13,22–25], but also in industrial IoT solutions (IETF 6TiSCH) [14], networks reliant on specific hardware and software environments [15,16], satellite communications [18], UAV networks [21], and infrastructures requiring hierarchical router architectures [19,22,23]. These findings underscore the relevance of adaptive queue scheduling methods across a wide range of scenarios and network environments.

Several key trends are evident in modern approaches to this field (see Table 2). The use of machine learning methods, particularly reinforcement learning (RL, DRL), allows adaptation to changing traffic conditions. However, they depend on computing resources and the quality of training data. Examples include algorithms for dynamic bandwidth allocation in WFQ [20], optimization of scheduling in UAV networks (ACBS-RL) [21], and scheduling functions for 6TiSCH [14].

Table 2. Comparative Analysis of Novel Approaches to Queue Management in Networking.

Ref.	Main Contribution	Advantages	Limitations
[13]	Proposed SW-EDF, a single-iteration switching algorithm for Combined Input-Output Queued (CIOQ) switches.	Reduces complexity to a single iteration, near Output-Queued performance, and low delay.	Only an approximation of stable matching, CIOQ requires costly stable matching.
[14]	Developed a scheduling function for IETF 6TiSCH networks based on multiweight evaluation and Q-Learning.	Precise slot selection, efficient resource use, less congestion/interference; improves latency by up to 55%, increases packet delivery ratio by 6%.	High computational cost, depends on training with real data (global network state).
[15]	Proposed a dynamic traffic scheduling algorithm for Huawei devices using priority queues.	Better resource utilization, bandwidth allocation, reduced delay and congestion, real-time adaptation.	Tested in simulations, limited to Huawei devices.
[16]	XDQ—an XDP/eBPF extension for programmable packet scheduling in Linux OS.	High performance, allows programmable scheduling within eXpress Data Path (XDP), reduces overhead.	XDQ still under development, requires Linux integration.
[17]	Opportunistic Weighted Fair Queueing (OWFQ)—approximate WFQ with calendar queues to reduce packet drops.	Significantly reduces packet drop rate, improves bandwidth utilization.	Approximation of ideal WFQ, validation limited to simulation environment.
[18]	QoS optimization for satellite-borne routers with multi-priority queues.	Simplifies optimization model, optimizes traffic arrival rates, achieves 30% QoS gain, fast convergence.	Limited to satellite-borne routers.
[19]	Multi-resource fair queueing algorithms with hierarchical scheduling—collapsed Hierarchical Dominant Resource Fair Queueing (collapsed H-DRFQ) and dove-tailing H-DRFQ.	Provides hierarchical QoS guarantees to individual flows, lower delay with dove-tailing H-DRFQ.	Validated only on the Click modular router.
[20]	WFQ-DRL algorithm—applied DRL to dynamic bandwidth allocation in WFQ.	Reduces delay and packet loss, adapts to real traffic.	Complex G/G/1/K model, limited scenarios tested.
[21]	Adaptive Credit-Based Shaper with Reinforcement Learning (ACBS-RL) method for UAV networks based on Static Priority-based Multiple Access (SPMA).	Ensures the QoS requirements for all priority traffic, increases UAV network throughput.	Relies on Q-learning-based reinforcement learning adaptation.
[22]	Hierarchical queue management on routers based on the goal coordination principle.	Scalable, balanced interface bandwidth allocation, fast convergence.	Theoretical approach, requires further real-world validation.
[23]	Two-level hierarchical queue management using interaction prediction principle.	High scalability, balanced and priority-based packet flow distribution, balanced bandwidth allocation, high convergence.	Limited experimental validation, hardware implementation challenges.
[24]	Developed an Active Queue Management method for network routers.	Reduced average packet delay, reduced packet loss, validated on Cisco lab setup.	Recommended mainly for high-load (>80%) scenarios, tested in the laboratory environment only.

Programmable solutions and extensions of existing platforms are becoming increasingly popular. Works [16,23] demonstrate that development is also moving towards the creation of flexible, programmable queue management mechanisms. At the same time, studies [19,22,23] focus on multi-level queue management to ensure QoS in complex networks with different classes of service. Such approaches allow balancing the load and ensuring fair resource allocation between flows. Also, some network environments need the latest specialized solutions. This applies, for example, to satellite networks [18] or Industrial IoT [14], where the algorithms are designed for specific environment requirements (low latency, energy efficiency, limited resources, etc.).

The approach to queue management based on the concept of Traffic Engineering Queues [20–26] deserves special attention. This concept applies the principle of load balancing across the network and communication links to optimize the utilization of available link and buffer resources, prevent overload, and improve QoS. The rationale behind this approach is that the leading QoS indicators (average delay, jitter, and packet loss probability) are functions of the utilization coefficients of network elements (interfaces, links, routes) and of the network as a whole [26–28].

For example, Ref. [24] proposes optimization-based mathematical models for solving Traffic Engineering Queue problems. In these models, flow distribution and interface bandwidth are balanced across a set of queues, taking into account the number of packets, their arrival rates, and the classes or priorities assigned to them. Such an approach corresponds to the requirements of DiffServ-TE technology. In studies [22,23], the model was refined and extended to support the implementation of hierarchical queues. Building on these results, the present work further develops the solutions proposed in [22–24] with a particular focus on ensuring QoS guarantees.

Consequently, the approaches discussed share the common objective of improving network resource utilization, reducing delays and packet loss, and ensuring QoS across different traffic classes. At the same time, most existing studies face certain limitations, including high implementation complexity and limited experimental validation. As a result, the development of queue management mechanisms is increasingly oriented toward adaptability, programmability, hierarchical structures, and specialization for particular network environments. While these directions enhance the efficiency of queue scheduling, they also pose challenges related to solution unification and ensuring practical scalability.

4. Class-Based Traffic Engineering Queue Model

First, the Class-Based Traffic Engineering Queue (CB-TEQ) model [24] is introduced as the basic solution, against which the enhanced models of the Traffic Engineering Queue family will be compared. Within this framework, the resource allocation problem must be addressed. We assume that packet flows arriving at the network interface, as specified by the routing tables, are already pre-aggregated and directed into one of several class queues. The criterion for assigning a given packet flow to a particular queue is the proximity of its class values [24].

The following notations are used:

B —router interface bandwidth, measured in bits per second;

N —number of class queues created on the interface;

b_i —interface bandwidth allocated for servicing the i th class queue;

r_i —average rate (intensity) of the aggregated packet flow arriving at the i th queue, measured in bits per second, where $i = \overline{1, N}$.

The class queue number i must be directly linked to the supported policies and methods for classification, marking (prioritization), and packet processing, such as DSCP/PHB in IP networks [12]. Ideally, each class queue can correspond to its own PHB policy or

decimal DSCP priority value. If the number of class queues is smaller than the number of policies or priorities, flows with similar DSCP/PHB values are typically combined (aggregated) into common class queues. A class queue with a lower index should serve lower-priority packet flows (i.e., those with lower Quality of Service requirements). In contrast, the queue with the highest index receives packets from the highest-priority flows, i.e., those with the highest QoS requirements.

From a physical standpoint, the variables b_i are subject to the following constraint:

$$b_i \geq 0, \quad (1)$$

$$\sum_{i=1}^N b_i = B. \quad (2)$$

Compliance with condition (2) ensures that there is no overload of the interface as a whole. Additionally, preventive conditions have been introduced for each class queue to prevent overload relative to the allocated bandwidth. These conditions ensure both the optimal resource allocation of interface bandwidth across queues and balanced utilization within the framework of the Traffic Engineering Queues concept [24]:

$$h_i^\alpha r_i \leq \alpha b_i \quad (i = \overline{1, N}), \quad (3)$$

where α is the control variable, defined as the upper dynamically regulated bound on queue utilization by bandwidth, subject to the following condition:

$$0 < \alpha \leq 1. \quad (4)$$

In turn, h_i^α is the class coefficient introduced to achieve a balanced allocation of interface bandwidth for the i th queue, taking into account its class value:

$$h_i^\alpha = 1 + \frac{i}{N \cdot D} \quad (i = \overline{1, N}), \quad (5)$$

Here, $D > 0$ denotes the normalization coefficient, which determines both the impact of the queue class on the coefficient h_i^α and the degree of bandwidth balancing across queues. As shown in the works of [23,24], the minimum allowable normalization coefficient D_{min} ensures the maximum level of service differentiation among packets in different class queues.

A higher queue class corresponds to a larger value of h_i^α . Consequently, the greater the class coefficient h_i^α , the lower the utilization ρ_i for the same boundary value α . Within the notation of model (1)–(5), the utilization coefficient is defined by the following expression:

$$\rho_i = \frac{r_i}{b_i} \quad (i = \overline{1, N}). \quad (6)$$

Moreover, an increase in the normalization coefficient D reduces the impact of the queue class on the allocated bandwidth volume. By introducing expressions (3)–(6), the model enables differentiated allocation of router interface bandwidth among the class queues. To achieve maximum differentiation in packet servicing across queues, the normalization coefficient D should be chosen as the minimum permissible value [23,24].

To ensure an effective and balanced distribution of interface bandwidth among queues, it is necessary to solve a nonlinear optimization problem with the optimality criterion $\min_{b, \alpha} \alpha$ and constraints (1)–(4). The physical interpretation of this criterion is to minimize the upper bound of queue utilization for each class. This approach ensures a balanced allocation of link resources across queues, in accordance with the principles of Traffic Engineering.

As a result, none of the queues becomes overloaded, and each queue receives an appropriate portion of the available interface bandwidth. A queue associated with a higher class value has a lower utilization coefficient, whereas a queue corresponding to the minimum class value exhibits the highest utilization coefficient. The main disadvantage of this approach lies in the increased computational complexity due to the nonlinear nature of constraint (3).

The nonlinearity of (3) follows from a bilinear term on its right-hand side, namely the product of the control variables b_i and α . In this formulation, all parameters on the left-hand side of (3) are known. The boundary value α ensures the balancing of the bandwidth required for servicing. Condition (3), therefore, illustrates the functional relationship among the control variables in the calculation process.

To transform (3) into a linear form, it can be rewritten as follows [24]:

$$\beta h_i^\alpha r_i \leq b_i \quad (i = \overline{1, N}), \quad (7)$$

where β is an additional control variable, defined as inversely proportional to the upper bound of interface queue utilization α , i.e.,

$$\beta = \frac{1}{\alpha}. \quad (8)$$

The following constraints are imposed on this variable:

$$\beta > 1. \quad (9)$$

The optimality criterion in queue management is defined by the following expression:

$$\beta \rightarrow \max. \quad (10)$$

According to expression (8), this criterion minimizes the upper bound of queue utilization α , weighted by the class values (3) and (5). Accordingly, the CB-TEQ solution (1)–(10) is formulated as a linear programming problem, with a linear optimality criterion (10) and a set of constraints (1), (2), (7), and (9). The linear formulation of the optimization problem has a favorable effect on the computational complexity of obtaining final solutions.

5. Enhancement of the Class-Based Traffic Engineering Queue Model

The CB-TEQ model fully meets the DiffServ-TE technology requirements for providing differentiated quality of service by balancing both the load and the interface bandwidth among class queues. To further ensure QoS guarantees consistent with IntServ conditions and to support the functionality of the Resource Reservation Protocol (RSVP) on network router interfaces, model (1)–(10) is extended by introducing the following conditions:

$$b_i^{req} 1_{(0, \infty)}(r_i) \leq b_i \quad (i = \overline{1, N}), \quad (11)$$

where b_i^{req} denotes the bandwidth requirements for ensuring QoS for packets in the i th queue, and $1_{(0, \infty)}(r_i)$ is the indicator function of the interval $(0, \infty)$:

$$1_{(0, \infty)}(r_i) = \begin{cases} 1 & \text{if } r_i > 0; \\ 0 & \text{if } r_i = 0. \end{cases} \quad (12)$$

The physical interpretation of condition (11) is that each class queue is guaranteed a certain portion of the interface bandwidth b_i^{req} , $i = \overline{1, N}$, but only if packets are present in that queue. Otherwise, when the queue is empty, no bandwidth is allocated to it. If

guaranteed bandwidth is not required for flows belonging to the j th class queue, then $b_j^{req} = 0$. In fact, conditions (11) and (12) serve as substitutes for conditions (1). The resulting formulation, associated with the minimization problem (10) and the updated set of constraints (2), (7), (9), and (11), is referred to as the Guarantee-Based Bandwidth Traffic Engineering Queue (GB(Bw)-TEQ).

Providing QoS guarantees by reserving a fixed portion of bandwidth for a particular class queue is a widely used approach in practice. However, bandwidth represents only one of the key QoS indicators. Most modern service flows are equally sensitive to other metrics, such as average delay, jitter, and packet loss probability [27]. It has been established that time QoS indicators and reliability measures are functions of the available bandwidth, or more specifically, of the utilization coefficient (6). For instance, as the utilization of the interface or an individual queue increases, both delay and packet loss grow correspondingly. To address this, the present work further proposes enhancing the CB-TEQ model by introducing guarantees for the utilization (6) for each class queue:

$$\rho_i \leq \rho_i^{req} \quad (i = \overline{1, N}), \quad (13)$$

where ρ_i^{req} denotes the QoS requirement for the utilization coefficient of the i th queue.

Taking into account expression (6), condition (13) can be rewritten in the following form:

$$\frac{r_i}{\rho_i^{req}} \leq b_i \quad (i = \overline{1, N}). \quad (14)$$

Condition (14) is analogous to (11), except that the value of b_i^{req} is now determined with respect to the actual load on a specific queue and the requirements for its utilization level, which provides a more adequate solution. This emphasizes improvements in time QoS indicators and packet delivery reliability. The resulting formulation, associated with the minimization (10) and the constraints (2), (7), (9), and (14), is referred to as the Guarantee-Based Utilization Traffic Engineering Queue (GB(U)-TEQ).

6. Investigation and Comparative Analysis of the Proposed Solutions

This study investigates and compares the CB-TEQ, GB(Bw)-TEQ, and GB(U)-TEQ solutions. The objective was to evaluate queue metrics on the router interface in terms of their compliance with DiffServ and IntServ requirements under different network load scenarios. In the experiments, the total number of class queues (service classes) was set to eight ($N = 8$). This choice ensured, on the one hand, a sufficient degree of service differentiation and, on the other, the clarity of the calculated results. In addition, this number of queues also corresponds to the typical number of IP priorities used for packet classification [6]. Numerical simulations were conducted in the MATLAB R2025a environment to evaluate the effectiveness of the proposed solutions.

The router interface bandwidth was fixed at 100 Mbps. Table 3 presents three scenarios of initial load distribution across the service flows of class queues. In Scenario I, the total incoming load on the interface R was 75 Mbps, while in Scenarios II and III, the loads were 80 Mbps and 90 Mbps, respectively. The characteristics of the first, sixth, and eighth flows remained unchanged across all scenarios to facilitate comparative analysis. These flows and their corresponding class queues are highlighted in gray in Table 3 (and in subsequent tables). The intensities of the remaining five aggregated flows, representing background traffic, varied randomly between scenarios.

Table 3. Initial Load Scenarios for Class Queues (Mbps).

Queue	Scenario I	Scenario II	Scenario III
1	12	12	12
2	8	10	8
3	6	8	7
4	14	15	18
5	7	8	15
6	11	11	11
7	5	4	7
8	12	12	12

In the study, the following class queue metrics ($i = \overline{1, N}$) were calculated and compared:

- allocated bandwidth (b_i , Mbps),
- utilization (ρ_i),
- average queue length (L_i).

Without loss of generality, the average queue length was computed using the formula corresponding to the simulation of a class queue as an M/M/1 system [28–33]:

$$L_i = \frac{\rho_i^2}{1 - \rho_i} \quad (i = \overline{1, N}). \quad (15)$$

The analysis aimed to evaluate and estimate the impact of the selected queue management model (CB-TEQ, GB(Bw)-TEQ, or GB(U)-TEQ), traffic load characteristics, and interface resource reservation schemes on these metrics.

Table 4 presents the results obtained for the CB-TEQ model. This data demonstrates that the allocation of interface bandwidth among individual queues is balanced according to class values: the higher the class, the lower the utilization and, correspondingly, the average queue length. This is evident from a comparison of the first and eighth class queues. The input flow intensities for both queues were intentionally set equal (12 Mbps each). However, under the CB-TEQ model, the higher-class queue received more bandwidth, which resulted in lower utilization and a shorter average length (Table 4). Thus, this approach fully complies with DiffServ requirements by providing a higher service quality to flows of higher classes compared to those of lower classes. At the same time, queues of the lowest class are not entirely deprived of service: they still receive a portion of the interface bandwidth, though they operate with the highest utilization coefficient.

Table 4. Queue Metrics for the CB-TEQ Model.

Queue	Scenario I ($D_{min}=2$)			Scenario II ($D_{min}=3$)			Scenario III ($D_{min}=6$)		
	b	ρ	L	b	ρ	L	b	ρ	L
1	13.30	0.9024	8.34	13.24	0.9064	8.78	12.44	0.9647	26.36
2	9.39	0.8522	4.91	11.47	0.8715	5.91	8.46	0.9454	16.37
3	7.43	0.8074	3.38	9.53	0.8393	4.38	7.55	0.9269	11.75
4	18.25	0.7670	2.52	18.54	0.8093	3.43	19.80	0.9090	9.08
5	9.58	0.7305	1.98	10.24	0.7814	2.79	16.82	0.8919	7.36
6	15.78	0.6973	1.61	14.56	0.7553	2.33	12.57	0.8754	6.15
7	7.50	0.6669	1.34	5.47	0.7310	1.99	8.14	0.8594	5.26
8	18.77	0.6392	1.13	16.95	0.7081	1.72	14.22	0.8441	4.57

Figure 3 illustrates the dependence of class queue utilization on the normalization coefficient D under the CB-TEQ model for Scenario I of the initial data (Table 4). The analysis of these dependencies (Figure 3) indicates that by adjusting D , the level of packet

service differentiation among class queues can be controlled. As D increases, the degree of differentiation decreases; at sufficiently high values, differentiation disappears altogether, making the creation of multiple queues on the interface irrelevant. Therefore, it is necessary to identify the minimum value of the normalization coefficient D_{min} that ensures maximum differentiation in packet servicing, which constitutes the primary purpose of organizing queues on the router interface.

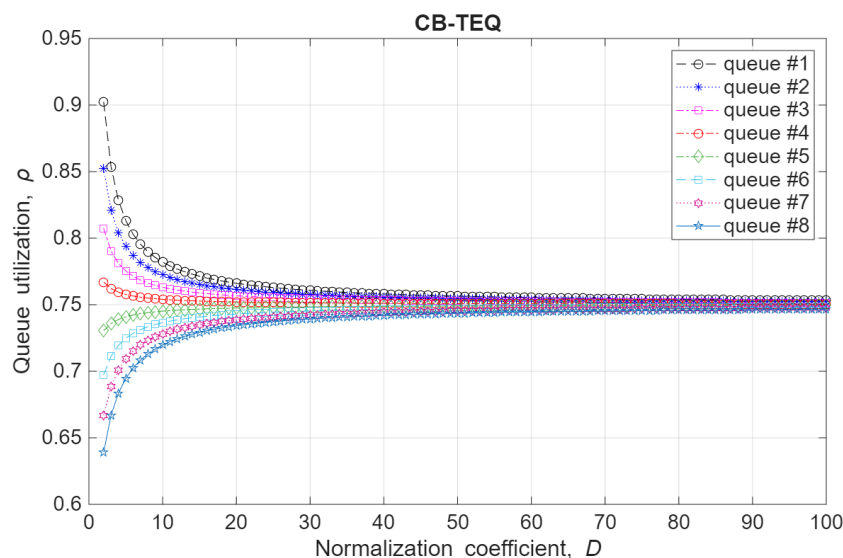


Figure 3. Relationship between queue utilization and the normalization coefficient D in the CB-TEQ model (Scenario I).

Experimental results indicate that the value of D_{min} is most sensitive to interface utilization. As interface utilization increases, D_{min} also increases (Table 4). In contrast, the number of configured and actively used class queues, as well as the structure of incoming traffic (the bitrate of aggregated packet flows), has a considerably smaller effect on D_{min} . It is recommended that when the interface utilization is between 50% and 65%, D_{min} be set between 0.6 and 1; when the utilization is between 65% and 90%, D_{min} should range from 1 to 5; and when the utilization is between 90% and 98%, D_{min} should range from 5 to 28.

The main functional limitation of the CB-TEQ solution is that it does not guarantee a specific service level for any flow or queue. As the interface load changes, as shown in Table 4, both the utilization factor and the average length of even high-class queues may fluctuate over a relatively wide range. This variability also affects QoS metrics such as average delay and packet loss probability.

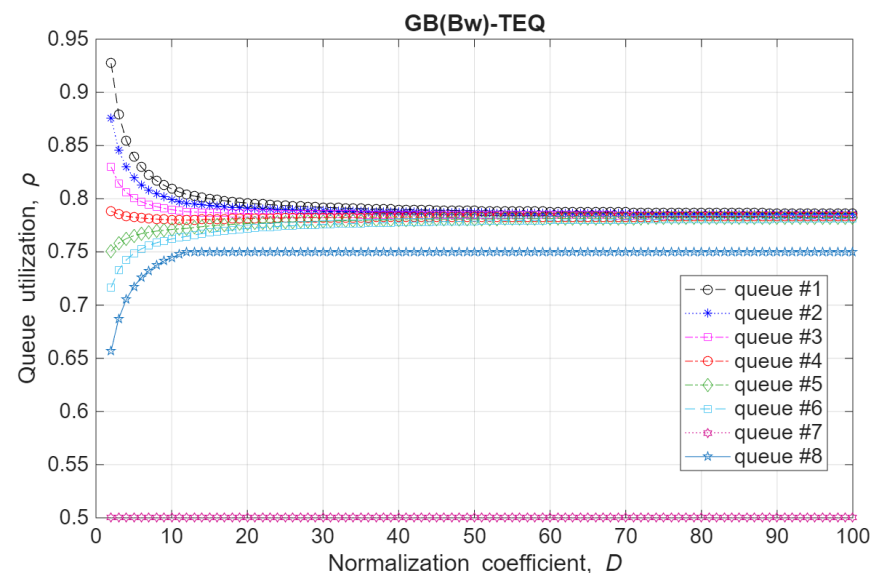
The results obtained for the GB(Bw)-TEQ model are presented in Table 5. In this case, it was assumed that the seventh and eighth queues required bandwidth reservations of 10 Mbps and 16 Mbps, respectively. That is, in expression (11), $b_7^{req} = 10$ Mbps and $b_8^{req} = 16$ Mbps. For the remaining queues, no bandwidth was reserved, i.e., $b_1^{req} = b_2^{req} = b_3^{req} = b_4^{req} = b_5^{req} = b_6^{req} = 0$. Based on the results presented in Table 5, the following conclusions can be drawn:

1. Irrespective of the load applied to the class queues, the seventh and eighth queues were guaranteed the specified portion of the interface bandwidth.
2. The guarantees referred to the minimum allocated bandwidth, meaning that class queues could receive additional resources. For instance, in Table 5, the eighth queue was allocated 18.27 Mbps under Scenario I and 16.13 Mbps under Scenario II.
3. The remaining interface bandwidth, after the reservation procedure, was distributed in a balanced manner among the queues according to their class values, as in the CB-TEQ case.

Table 5. Queue Metrics for the GB(Bw)-TEQ Model.

Queue	Scenario I ($D_{min}=2$)			Scenario II ($D_{min}=3$)			Scenario III ($D_{min}=11$)		
	b	ρ	L	b	ρ	L	b	ρ	L
1	12.94	0.9275	11.86	12.60	0.9520	18.88	12.14	0.9885	84.67
2	9.13	0.8759	6.18	10.92	0.9154	9.90	8.18	0.9775	42.42
3	7.23	0.8298	4.05	9.08	0.8815	6.56	7.24	0.9667	28.10
4	17.76	0.7883	2.94	17.65	0.8500	4.82	18.83	0.9562	20.89
5	9.32	0.7508	2.26	9.75	0.8207	3.76	15.86	0.9459	16.55
6	15.35	0.7167	1.81	13.87	0.7933	3.05	11.75	0.9359	13.66
7	10.00	0.5000	0.50	10.00	0.4000	0.27	10.00	0.7000	1.63
8	18.27	0.6569	1.26	16.13	0.7438	2.16	16.00	0.7500	2.25

Figure 4 presents the dependence of class queue utilization on the normalization coefficient D under the GB(Bw)-TEQ model, also for Scenario I of the initial data (Table 5). A comparison of Figures 3 and 4 indicates that providing QoS guarantees in terms of bandwidth within the GB(Bw)-TEQ model influences the nature of the observed dependencies. Since $b_7^{req} = 10$ Mbps, $b_8^{req} = 16$ Mbps, $r_7 = 5$ Mbps, and $r_8 = 12$ Mbps, the maximum utilization coefficients of these queues should not exceed 0.5 and 0.75, respectively (Figure 4), as the normalization coefficient D varies.

**Figure 4.** Relationship between queue utilization and the normalization coefficient D in the GB(Bw)-TEQ model (Scenario I).

A limitation of the GB(Bw)-TEQ solution is that bandwidth is reserved for the selected queues regardless of their actual load. This can be observed in Table 5, for example, in the case of the seventh queue, which may result in inefficient (excessive) bandwidth allocation, particularly when the queue with reserved bandwidth experiences a reduced load.

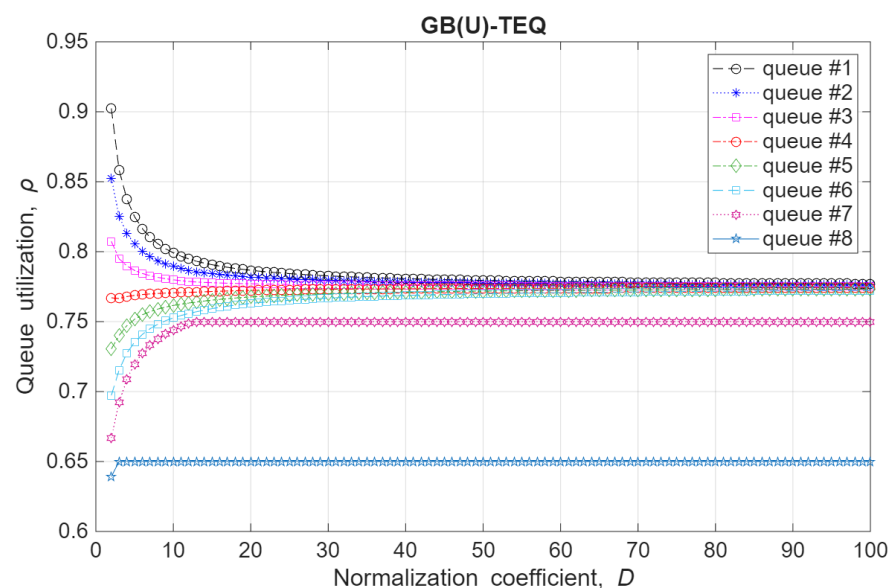
The results obtained for the GB(U)-TEQ model are presented in Table 6. In this case, for the seventh and eighth queues, it was necessary to guarantee utilization coefficients not exceeding 0.75 and 0.65, respectively. That is, in expression (11), $\rho_7^{req} = 0.75$ and $\rho_8^{req} = 0.65$. No such guarantees were applied to the other queues. The analysis of Table 6 leads to the following conclusions:

Table 6. Queue Metrics for the GB(U)-TEQ Model.

Queue	Scenario I ($D_{min}=2$)			Scenario II ($D_{min}=3$)			Scenario III ($D_{min}=28$)		
	b	ρ	L	b	ρ	L	b	ρ	L
1	13.30	0.9024	8.34	13.00	0.9232	11.11	12.05	0.9951	200.69
2	9.39	0.8522	4.91	11.26	0.8877	7.02	8.08	0.9907	105.11
3	7.43	0.8074	3.38	9.36	0.8549	5.03	7.10	0.9863	71.00
4	18.25	0.7670	2.52	18.20	0.8243	3.87	18.33	0.9820	53.49
5	9.58	0.7305	1.98	10.05	0.7959	3.10	15.34	0.9777	42.84
6	15.78	0.6973	1.61	14.30	0.7694	2.57	11.30	0.9734	35.67
7	7.50	0.6670	1.34	5.37	0.7446	2.17	9.33	0.7500	2.25
8	18.77	0.6392	1.13	18.46	0.6500	1.21	18.46	0.6500	1.21

1. In accordance with the values of ρ_7^{req} and ρ_8^{req} , and depending on the load received by the class queues, the seventh and eighth queues were guaranteed a specified portion of interface bandwidth.
2. The guarantees applied to the maximum allowable utilization coefficient of the selected queues, meaning that they could receive additional link resources. For example, in Table 6, the eighth queue under Scenario I had $\rho_8 = 0.6392$, which was below ρ_8^{req} , and the seventh queue under Scenarios I and II had $\rho_7 = 0.667$ and 0.7446 , respectively, both below the threshold ρ_7^{req} .
3. The interface bandwidth remaining after the reservation procedure was distributed evenly among the queues according to their class values, as in the CB-TEQ case.

Figure 5 presents the dependence of class queue utilization coefficients on the normalization coefficient D under the GB(U)-TEQ model, again for Scenario I of the initial data (Table 6). These dependencies follow the same logic as those shown in Figure 4. However, in this case, the upper bounds of the utilization coefficients for the seventh and eighth queues were defined by the initial requirements, namely $\rho_7^{req} = 0.75$ and $\rho_8^{req} = 0.65$. They must not be exceeded either when the load on the interface and individual queues changes or when the normalization coefficient D varies.

**Figure 5.** Relationship between queue utilization and the normalization coefficient D in the GB(U)-TEQ model (Scenario I).

Therefore, the GB(U)-TEQ solution is more adaptive than CB-TEQ and GB(Bw)-TEQ in terms of considering flow/queue classes and changes in load characteristics (number of

flows and their intensities). Guarantees on the maximum allowable utilization coefficient for selected class queues have a positive influence on the stabilization of average queue length and, consequently, on the control of average packet delay and loss rate. However, if additional constraints are imposed on the limit values of delay and/or packet loss for a particular class queue, the GB-TEQ model will lose its linearity. This results from the fact that the analytical relationships linking QoS indicators, queue characteristics, and flows are inherently nonlinear [28].

Figures 3–5 show the results of the study for Scenario I only. This is due to the following conclusion: the general nature of the dependencies $\rho(D)$ remained unchanged for different scenarios (I, II, and III). The value of D_{min} , as indicated in Tables 4–6, has changed. Scenario I was chosen for demonstration because the lines corresponding to the seventh and eighth queues present noticeable curvature, as clearly shown in Figure 5. For these queues, the dependencies $\rho(D)$ first increase and then stabilize at the level of bandwidth or utilization requirements. If the interface utilization is higher (scenarios II and III), a situation similar to Figure 4 will be observed: the dependence $\rho(D)$ for the seventh queue is trivial and represents a straight line.

7. Practical Recommendations for Applying the Proposed Solutions

The GB(Bw)-TEQ and GB(U)-TEQ solutions proposed in this work can serve as a foundation for mathematical and algorithmic software for switches and routers in both traditional IP/MPLS networks and software-defined networks (SDN). These solutions enable adaptation to network state changes and efficient allocation of interface bandwidth among class queues and, ultimately, among packet flows to ensure guaranteed and differentiated service (Figure 6). Such adaptation may involve monitoring the characteristics of the traffic arriving at the interface by analyzing and responding to variations in the number of flows and their attributes, such as average intensity, priority, packet length, and other parameters that typically affect packet classification. Based on the results of this classification, a set of class queues can be organized on the interface, with link resources allocated among them using the GB(Bw)-TEQ or GB(U)-TEQ models. The application of these models enables nearly automated queue management and reduces reliance on routine administrative configurations still required in PQ, CQ, CBQ, CBWFQ, and LLQ mechanisms (Table 1).

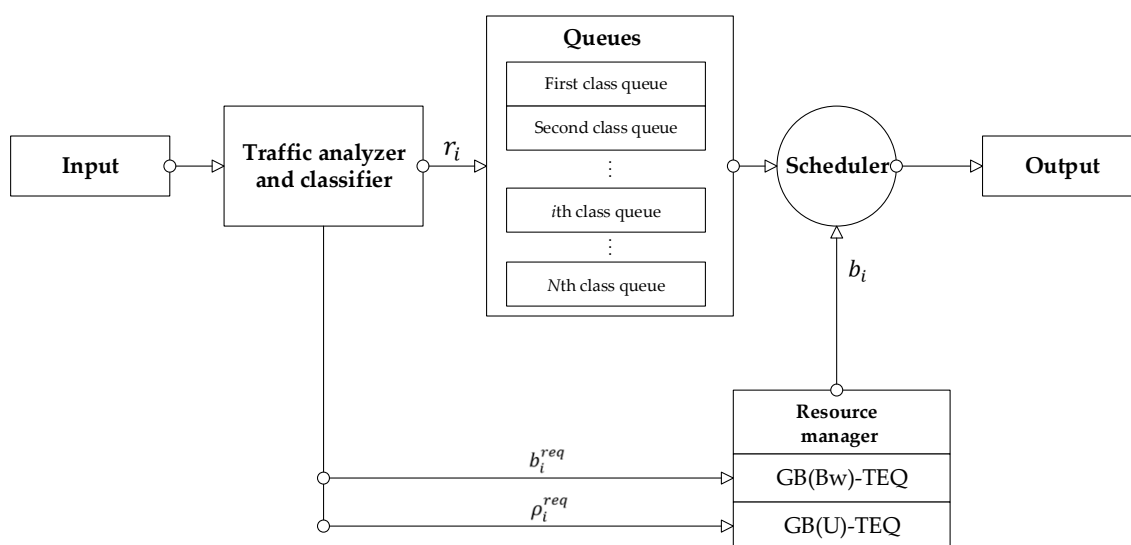


Figure 6. Queue management architecture on a router interface according to the GB(Bw)-TEQ and GB(U)-TEQ models.

8. Conclusions

This work proposes a system of solutions for managing queues on network router interfaces, represented by the GB(Bw)-TEQ and GB(U)-TEQ models. These solutions conform to the principles of the Traffic Engineering Queues concept and support the implementation of both DiffServ and IntServ. The novelty of the GB(Bw)-TEQ and GB(U)-TEQ models lies in their ability to provide guarantees either for the bandwidth allocated to a given class queue or for its utilization coefficient. Such guarantees stabilize and control the average queue length, which in turn positively affects QoS indicators such as average delay and packet loss probability. The unreserved portion of the interface bandwidth is distributed evenly among packet queues according to their class values. As a result, higher-class queues show lower utilization, while lower-class queues experience higher utilization, which is consistent with the principles of DiffServ.

The proposed GB(Bw)-TEQ and GB(U)-TEQ models are formulated as mixed-integer linear programming (MILP) optimization problems with an optimality criterion (10) and a set of constraints (2), (7), (9), and either (11) or (14). When guarantees are expressed in terms of bandwidth allocated to a class queue (GB(Bw)-TEQ), constraint (11) is applied. When guarantees are defined in terms of utilization level (GB(U)-TEQ), constraint (14) is used instead of (11).

A series of numerical examples confirmed the feasibility and effectiveness of the proposed GB(Bw)-TEQ and GB(U)-TEQ models by comparing them with the previously known CB-TEQ analogue [24], which did not provide service-level guarantees. The results validated the properties of the developed optimization models in terms of both providing the stated guarantees and enabling differentiated packet service according to queue class values. Importantly, balancing interface bandwidth among queues did not result in the blocking of the lowest-class queues, a drawback often observed in mechanisms such as PQ or CQ. The solutions proposed in this work can be applied to automate queue management in switches and routers for both traditional IP/MPLS networks and software-defined networks.

Author Contributions: Conceptualization, L.T., O.L., O.Y., R.S. and A.B.; methodology, O.L., O.Y. and R.S.; software, O.L., O.Y. and R.S.; validation, L.T., O.L. and A.B.; formal analysis, L.T., O.L., O.Y., R.S. and A.B.; investigation, O.L., O.Y. and R.S.; resources, O.L., O.Y. and R.S.; data curation, L.T. and A.B.; writing—original draft preparation, L.T., O.L., O.Y., R.S. and A.B.; writing—review and editing, L.T., O.L., O.Y., R.S. and A.B.; visualization, O.L., O.Y. and R.S.; supervision, L.T., O.L., O.Y., R.S. and A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data supporting this study are included within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CBQ	Class-based Queuing
CB-TEQ	Class-Based Traffic Engineering Queue
CQ	Custom Queuing
DiffServ	Differentiated Services
DRL	Deep Reinforcement Learning
DSCP	Differentiated Services Code Point
FIFO	First In, First Out
GB(Bw)-TEQ	Guarantee-Based Bandwidth Traffic Engineering Queue

GB(U)-TEQ	Guarantee-Based Utilization Traffic Engineering Queue
IntServ	Integrated Services
IP	Internet Protocol
LLQ	Low Latency Queueing
MILP	Mixed-Integer Linear Programming
PHB	Per-Hop Behavior
PQ	Priority Queuing
QoS	Quality of Service
RL	Reinforcement Learning
RSVP	Resource Reservation Protocol
TE	Traffic Engineering
WFQ	Weighted Fair Queueing

References

1. Troia, S.; Borgianni, L.; Sguotti, G.; Giordano, S.; Maier, G. A Comprehensive Survey on Software-Defined Wide Area Network. *IEEE Commun. Surv. Tutor.* **2025**, 1–40, Early Access. [CrossRef]
2. Zambouri, K.; Noor-A-Rahim, M.; John, J.; Sreenan, C.J.; Poor, H.V.; Pesch, D.K. A Comprehensive Survey of Wireless Time-Sensitive Networking (TSN): Architecture, Technologies, Applications, and Open Issues. *IEEE Commun. Surv. Tutor.* **2025**, 27, 2129–2155. [CrossRef]
3. Cristobo, L.; Ibarrola, E.; Casado-O'Mara, I.; Zabala, L. Global Quality of Service (QoX) Management for Wireless Networks. *Electronics* **2024**, 13, 3113. [CrossRef]
4. Tache, M.D.; Păscuțoiu, O.; Borcoci, E. Optimization Algorithms in SDN: Routing, Load Balancing, and Delay Optimization. *Appl. Sci.* **2024**, 14, 5967. [CrossRef]
5. Ma, Z.; Zhang, R.; Ai, B.; Lian, Z.; Zeng, L.; Niyato, D. Deep Reinforcement Learning for Energy Efficiency Maximization in RSMA-IRS-Assisted ISAC System. 2025. *IEEE Trans. Veh. Technol.* 2025, Early Access. [CrossRef]
6. Relington, J. *QoS in IP Networks: Prioritization, Classification, and Traffic Shaping*, Kindle ed.; 2025; 225p. Available online: <https://www.amazon.com/QoS-IP-Networks-Prioritization-Classification/dp/B0F3W4JBCT> (accessed on 1 August 2025).
7. CISCO. *QoS: Congestion Management Configuration Guide*; Cisco IOS XE 17; Cisco Systems, Inc.: San Jose, CA, USA, 2019.
8. Barreiros, M.; Lundqvist, P. *QoS-Enabled Networks: Tools and Foundations*; John Wiley & Sons: Hoboken, NJ, USA, 2016; 256p.
9. Edgeworth, B.; Rios, R.G.; Gooley, J.; Hucaby, D. *CCNP and CCIE Enterprise Core ENCOR 350-401 Official Cert Guide*, 2nd ed.; Cisco Press: Hoboken, NJ, USA, 2023; 1072p.
10. Chahed, H.; Kassler, A. TSN Network Scheduling—Challenges and Approaches. *Network* **2023**, 3, 585–624. [CrossRef]
11. Bernet, Y.; Ford, P.; Yavatkar, R.; Baker, F.; Zhang, L.; Speer, M.; Braden, R.; Davie, B.; Wroclawski, J.; Felstaine, E. RFC2998: A Framework for Integrated Services Operation over Diffserv Networks. 2000. Available online: <https://www.rfc-editor.org/rfc/rfc2998.html> (accessed on 1 August 2025).
12. Grossman, D. RFC 3260: New Terminology and Clarifications for Diffserv. 2002. Available online: <https://www.rfc-editor.org/info/rfc3260> (accessed on 1 August 2025).
13. Yu, Q.; Meng, J.; Xu, J.J. SW-EDF: A Single-Iteration Algorithm for Combined Input- and Output-Queued Switching. In Proceedings of the 2025 IEEE 26th International Conference on High Performance Switching and Routing (HPSR), Suita, Japan, 20–22 May 2025; pp. 1–7. [CrossRef]
14. Yang, W.; Luo, H.; Luo, S.; Zhang, Z.; Wang, X.; Liu, T. Efficient Scheduling Function for IETF 6TiSCH Networks Based on Multiweight Evaluation and Improved Q-Learning. *IEEE Internet Things J.* **2025**, 12, 35731–35743. [CrossRef]
15. Lin, H.; Wang, H.; Wang, N.; Luo, D. Research on Dynamic Traffic Scheduling Algorithm Based on Huawei Network Equipment. In Proceedings of the 2024 IEEE 4th International Conference on Data Science and Computer Application (ICDSCA), Dalian, China, 22–24 November 2024; pp. 578–584. [CrossRef]
16. Alfredsson, F.; Hurtig, P.; Brunstrom, A.; Høiland-Jørgensen, T.; Brouer, J.D. XDQ: Enhancing XDP with Queuing and Packet Scheduling. In Proceedings of the 2024 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), Paris, France, 11–14 March 2024; pp. 52–56. [CrossRef]
17. Yu, X.; Chen, W.; Tian, Y. OWFQ: Reducing Packet Drops for Approximate Weighted Fair Queueing with Calendar Queues. In Proceedings of the 2023 9th International Conference on Computer and Communications (ICCC), Chengdu, China, 8–11 December 2023; pp. 540–544. [CrossRef]
18. Zhu, Y. Quality of Service Optimization for Satellite-Borne Router with Multi-Priority Scheduling Queues. *IEEE Commun. Lett.* **2023**, 27, 3003–3007. [CrossRef]

19. You, C.; Zhao, Y.; Feng, G.; Quek, T.Q.; Li, L. Hierarchical Multiresource Fair Queueing for Packet Processing. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 726–740. [[CrossRef](#)]
20. Pan, J.; Chen, G.; Wu, H.; Peng, X.; Xia, L. Deep Reinforcement Learning-based Dynamic Bandwidth Allocation in Weighted Fair Queues of Routers. In Proceedings of the 2022 IEEE 18th International Conference on Automation Science and Engineering (CASE), Mexico City, Mexico, 20–24 August 2022; pp. 1580–1587. [[CrossRef](#)]
21. Yang, K.; Xu, C.; Qiao, G.; Zhong, J.; Zhang, X. Intelligent Queue Scheduling Method for SPMA-Based UAV Networks. *Drones* **2025**, *9*, 552. [[CrossRef](#)]
22. Lemeshko, O.; Persikov, A.; Yeremenko, O.; Yevdokymenko, M. Method of Hierarchical Queue Management on Network Routers Based on the Goal Coordination Principle. In *Advanced Smart Information and Communication Technology and Systems. MCT 2024; Lecture Notes in Networks and Systems*; Springer: Cham, Switzerland, 2025; Volume 1470, pp. 200–214. [[CrossRef](#)]
23. Lemeshko, O.; Yeremenko, O.; Titarenko, L.; Barkalov, A. Hierarchical Queue Management Priority and Balancing Based Method under the Interaction Prediction Principle. *Electronics* **2023**, *12*, 675. [[CrossRef](#)]
24. Lemeshko, O.; Lebedenko, T.; Holoveshko, M. Development and Research of Active Queue Management Method on Interfaces of Telecommunication Networks Routers. In *Data-Centric Business and Applications; Lecture Notes on Data Engineering and Communications Technologies*; Springer: Cham, Switzerland, 2021; Volume 69, pp. 1–20. [[CrossRef](#)]
25. Lemeshko, O.; Yeremenko, O.; Persikov, A.; Yevdokymenko, M.; Mersni, A.; Harkusha, S. Queue Management Priority-Based Traffic Engineering Method. In Proceedings of the 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 10–12 October 2022; pp. 479–483. [[CrossRef](#)]
26. Wu, G. An Innovative Priority Queueing Strategy for Mitigating Traffic Congestion in Complex Networks. *Mathematics* **2025**, *13*, 495. [[CrossRef](#)]
27. ITU-T Rec. Y.1541. Network Performance Objective for IP-Based Services. December 2011. 66p. Available online: <https://www.itu.int/rec/T-REC-Y.1541-201112-I/en> (accessed on 1 September 2025).
28. Lakatos, L.; Szeidl, L.; Telek, M. *Introduction to Queueing Systems with Telecommunication Applications*, 2nd ed.; Springer: Cham, Switzerland, 2019; 568p. [[CrossRef](#)]
29. Hwang, R.H.; Lin, J.Y.; Lin, J.Y.; Lin, Y.D. Communication and Computing Offloading in 6G Hotspot Scenarios. In Proceedings of the 2025 IEEE International Conference on Communications Workshops (ICC Workshops), Montreal, QC, Canada, 8–12 June 2025; pp. 1525–1531. [[CrossRef](#)]
30. Kumar, M.S.; Dadlani, A.; Tabassum, H. Age of Information in Unreliable Tandem Queues. *IEEE Commun. Lett.* **2025**, *29*, 2308–2312. [[CrossRef](#)]
31. Chen, Z.; Zhu, J.; Pappas, N.; Zhang, L.; Wang, M.; Tang, C.; Quek, T.Q. Real-Time Status Update System in a Parallel Blocking Queue. *IEEE Open J. Commun. Soc.* **2025**, *6*, 4608–4623. [[CrossRef](#)]
32. Roy, A.; Pachua, J.L.; Singh, N.B.; Saha, A.K. Quantum inspired genetic algorithm and optimization of queueing delay. In Proceedings of the TENCON 2024-2024 IEEE Region 10 Conference (TENCON), Singapore, 1–4 December 2024; pp. 1929–1933. [[CrossRef](#)]
33. Lin, W.; Li, L.; Yuan, J.; Han, Z.; Juntti, M.; Matsumoto, T. Age-of-Information in First-Come-First-Served Wireless Communications: Upper Bound and Performance Optimization. *IEEE Trans. Veh. Technol.* **2022**, *71*, 9501–9515. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.